УДК 01.05.02; 05.13.06; 05.13.21

А. Ковальчук, А. Шевчук, В. Цепак
Національний університет "Львівська політехніка",
кафедра інформаційних технологій видавничої справи

ПОБІТОВІ ОПЕРАЦІЇ Й ЕЛЕМЕНТИ АЛГОРИТМУ RSA В ШИФРУВАННІ-ДЕШИФРУВАННІ КОЛЬОРОВИХ ЗОБРАЖЕНЬ

Описано поєднання елементів алгоритму RSA і побітових операцій для сумісного використання під час шифрування-дешифрування зображень. Шифрування-дешифрування здійснюють без додаткового зашумлення.

Ключові слова: шифрування, дешифрування, алгоритм RSA, побітова операція.

Described combination of elements of the RSA algorithm and a bit-wise operations for the joint use for encryption - interpretation of images. Encryption - decryption is performed without additional noise.

Key words: encryption, decryption, the RSA algorithm, bitwise operation.

Вступ

Сьогодні асиметричне шифрування на основі відкритого ключа RSA використовує більшість продуктів на ринку інформаційної безпеки. Проблемою симетричного шифрування є необхідність передавання ключа для розшифрування інформації. Отже, ключ може бути перехоплений кимось іншим. В асиметричному шифруванні є два пов'язані ключі, які співпрацюють у парі, тобто коли інформація шифрується відкритим ключем, то розшифровування відбувається тільки відповідним секретним ключем та навпаки. Такий метод допомагає одержувачеві контролювати цілісність інформації, яку передають.

Зображення – це відтворення предметів і явищ об'єктивної реальності променями, які пройшли оптичну систему з центрованих сферичних поверхонь, що мають одну загальну оптичну вісь. Ілюстрації, виконані засобами растрової графіки, рідко створюють вручну, за допомогою комп'ютерних програм. Найчастіше застосовують відскановані ілюстрації, підготовлені художником на папері, чи фотографії. Останнім часом для введення растрових зображень у комп'ютер широко застосовують цифрові фото- і відеокамери. Відповідно, більшість растрових графічних редакторів орієнтовані не стільки на створення зображень, скільки на їх обробку.

Серед усіх сенсорних систем зорова найінформативніша. За підрахунками вчених, близько 90 % усієї інформації ми отримуємо завдяки зору. Водночас зображення як стохастичний сигнал є одною із найуживаніших форм представлення інформації. Саме тому актуальним завданням є захист зображень від несанкціонованого доступу та використання.

Окрім вербальної інформативності, зображення володіє і візуальною інформативністю, яка уможливлює організацію несанкціонованого доступу для сучасних методів обробки зображень. Фактично організація атак на зашифровані зображення буває двох типів: класичний злам за допомогою методів шифрування і з використанням методів візуальної обробки зображень (злам можна здійснити за лічені години чи хвилини, а часом і в режимі реального часу). У зв'язку з цим перед методами шифрування у випадку їхнього застосування висувається ще одна вимога – повне шифрування шумів зображення. Це необхідно для запобігання використанню візуальних методів відтворення зображень.

Під час шифрування-дешифрування цифрових зображень алгоритмом RSA виникає проблема збереження контурів (флуктуації функції інтенсивності) на зашифрованому зображенні, що містить додаткову інформацію [3–5]. Контури – це ті області зображення, де виникають різкі зміни інтенсивності. Застосовуючи методи цифрової обробки зображення, можна отримати практично

повну інформацію про зашифроване зображення [1, 2]. Для вирішення цієї проблеми і використовують побітові операції в алгоритмі RSA із додатковим зашумленням в програмній реалізації. Крім того, алгоритм виокремлення контурів має відрізнятися якомога вищою швидкодією і при цьому створювати компактний опис контуру, придатний для подальшого аналізу.

Мета роботи

Відносно зображення актуальною задачею є таке використання класичного алгоритму RSA, щоб:

– не знизити криптографічної стійкості алгоритму RSA;

– забезпечити повну зашумленість зображення з метою унеможливити використання методів візуальної обробки зображень [6].

Одним із шляхів створення такої модифікації є поєднання елементів алгоритму RSA і побітових операцій у програмній реалізації.

Характеристики зображення

Нехай задано зображення $\boldsymbol{P}$ шириною $\boldsymbol{l}$ і висотою $\boldsymbol{h}$. Його можна розглядати як матрицю інтенсивностей пікселів

$$Z = \begin{pmatrix} z_{1,1} & \dots & z_{1,l} \\ \dots & \dots & \dots \\ z_{h,1} & \dots & z_{h,l} \end{pmatrix}, \quad (1)$$

де z_{ij} – значення інтенсивності піксела [1].

Задача виокремлення контура потребує використання операцій над сусідніми елементами, які є чутливими до змін і пригашають області постійних рівнів яскравості, тобто контури – це ті області, де виникають зміни, стаючи світлими, тоді як інші частини зображення залишаються темними. Тому виокремлення контура означає пошук найрізкіших змін, тобто максимумів модуля вектора градієнта [2]. Це є однією з причин, через що контури залишаються в зображенні при шифруванні в системі RSA, оскільки шифрування тут ґрунтується на піднесенні до степеня за модулем деякого натурального числа. При цьому на контурі і на сусідніх до контура пікселах піднесення до степеня значення яскравостей дає ще більший розрив.

Опис алгоритму шифрування.

Шифрування за двома рядками матриці зображення

Нехай $\boldsymbol{P, Q}$ – пара довільних простих чисел і $\boldsymbol{N = P * Q}$, $\boldsymbol{j(N) = (P - 1)(Q - 1)}$ – функція Ейлера**.** Шифрування відбувається поелементно з використанням такого перетворення елементів матриці зображення $\boldsymbol{Z}$:

1. За алгоритмом RSA вибирають числа $\boldsymbol{e < j(N)}$, $\boldsymbol{d < j(N)}$, що виконується конгруенція $\boldsymbol{ed \circ 1 (\mathrm{mod}\ j(N))}$.

2. Для $\boldsymbol{i}$-го рядка матриці, $\boldsymbol{1 \le i \le l}$, вибирається число $\boldsymbol{m \equiv (i + P)\ (\mathrm{mod}\ 32)}$ і будуються числа $\boldsymbol{A \circ m^e\ (\mathrm{mod}\ N)}$, $\boldsymbol{X = i*A*P}$.

3. Для $\boldsymbol{i}$ +1-го рядка матриці, $\boldsymbol{1 \le i \le l}$, вибирається число $\boldsymbol{n \equiv (i + Q)\ (\mathrm{mod}\ 32)}$ і будуються числа $\boldsymbol{B \circ n^d\ (\mathrm{mod}\ N)}$, $\boldsymbol{Y = i*B*Q}$.

4. З використанням бінарної операції ^ - порозрядного виключеного “АБО” - будуються числа $\boldsymbol{a = z_{i,j} \wedge X}$ та $\boldsymbol{b = z_{i+1,j} \wedge Y}$.

5. Виокремлюється кожний розряд $\boldsymbol{a_i}$ числа $\boldsymbol{a}$ за такою схемою: a_1= a & 01; a_2= a & 02; a_3 = =a & 04; a_4= a & 010; a_5 = a & 020; a_6 = a & 040; a_7 = a & 0100; a_8 = a & 0200; a_9 = a & 0400; a_{10} = =a & 01000; a_{11} = a & 02000; a_{12} = a & 04000; a_{13} = a & 010000; a_{14} = a & 020000; a_{15} = a & 040000; a_{16} = a & 0100000; a_{17} = a & 0200000; a_{18} = a & 0400000; a_{19} = a & 01000000; a_{20} = a & 02000000; a_{21} = = a & 04000000; a_{22} = a & 010000000; a_{23} = a & 020000000; a_{24} = a & 040000000; a_{25} = a & 0100000000; a_{26} = a & 0200000000; a_{27} = a & 0400000000; a_{28} = a & 01000000000; a_{29} = a & 02000000000; a_{30} = =a & 04000000000; a_{31} = a & 010000000000; a_{32} = a & 020000000000, де & – операція арифметичного “І”.

6. Виконується циклічне заміщення $m + 1$ розрядів числа a за схемою: $k = a_{m+1}$, $a_{m+1} = a_m$, …, $a_2 = a_1$, $a_1 = k$.

7. Виокремлюється кожний розряд b_i числа b за такою схемою: b_1= b & 01; b_2= b & 02; b_3 = b & 04; b_4= b & 010; b_5 = b & 020; b_6 = b & 040; b_7 = b & 0100; b_8 = b & 0200; b_9 = b & 0400; b_{10} = b & 01000; b_{11} = b & 02000; b_{12} = b & 04000; b_{13} = b & 010000; b_{14} = b & 020000; b_{15} = b & 040000; b_{16} = b & 0100000; b_{17} = b & 0200000; b_{18} = b & 0400000; b_{19} = b & 01000000; b_{20} = b & 02000000; b_{21} = b & 04000000; b_{22} = b & 010000000; b_{23} = b & 020000000; b_{24} = b & 040000000; b_{25} = b & 0100000000; b_{26} = = b & 0200000000; b_{27} = b & 0400000000; b_{28} = b & 01000000000; b_{29} = b & 02000000000; b_{30} = b & 04000000000; b_{31} = b & 010000000000; b_{32} = b & 020000000000, де & - операція арифметичного “І”.

8. Виконується циклічне заміщення $n + 1$ розрядів числа b за схемою: $k = b_{n+1}$, $b_{n+1} = b_n$, …, $b_2 = b_1$, $b_1 = k$.

9. Зашифрованим є зображення після кроків 5 – 8.

10. Всі отримані числа записуються в матрицю

$$V = \begin{pmatrix} v_{1,1} & \dots & v_{1,l} \\ \dots & \dots & \dots \\ v_{h,1} & \dots & v_{h,l} \end{pmatrix}.$$

Дешифрування за двома рядками матриці зображення

Дешифрування проводять при заданих числах $e < j(N)$ і d, $N = P * Q$, $j(N) = (P - 1)(Q - 1)$.

1. Для i-го рядка матриці, $1 \le i \le l$, вибирається число $m \equiv (i + P) \pmod{32}$ і будуються числа $A \circ m^e \pmod{N}$, $X = i*A*P$.

2. Для $i + 1$-го рядка матриці, $1 \le i \le l$, вибирається число $m \equiv (i + Q) \pmod{32}$ і будуються числа $B \circ m^d \pmod{N}$, $Y = i*B*Q$.

3. Виокремлюється кожний розряд a_i числа a за схемою: a_1 = a & 01; a_2 = a & 02; a_3 = a & 04; a_4 = a & 010; a_5 = a & 020; a_6 = a & 040; a_7 = a & 0100; a_8 = a & 0200; a_9 = a & 0400; a_{10} = a & 01000; a_{11} = a & 02000; a_{12} = a & 04000; a_{13} = a & 010000; a_{14} = a & 020000; a_{15} = a & 040000; a_{16} = a & 0100000; a_{17} = a & 0200000; a_{18} = a & 0400000; a_{19} = a & 01000000; a_{20} = a & 02000000; a_{21} = a & 04000000; a_{22} = = a & 010000000; a_{23} = a & 020000000; a_{24} = a & 040000000; a_{25} = a & 0100000000; a_{26} = =a & 0200000000; a_{27} = a & 0400000000; a_{28} = a & 01000000000; a_{29} = a & 02000000000; a_{30} = =a & 04000000000; a_{31} = a & 010000000000; a_{32} = a & 020000000000, де & - операція арифметичного “І”.

4. Виконується циклічне заміщення $m + 1$ розрядів числа a за схемою: $k = a_{m+1}$, $a_{m+1} = a_m$, …, $a_2 = a_1$, $a_1 = k$.

5. З використанням бінарної операції ^ - порозрядного виключеного “АБО” – будується число $z_{i,j} = a \wedge X$.

6. Виокремлюється кожний розряд b_i числа b за такою схемою: b_1= b & 01; b_2= b & 02; b_3 = =b & 04; b_4= b & 010; b_5 = b & 020; b_6 = b & 040; b_7 = b & 0100; b_8 = b & 0200; b_9 = b & 0400; b_{10} = =b & 01000; b_{11} = b & 02000; b_{12} = b & 04000; b_{13} = b & 010000; b_{14} = b & 020000; b_{15} = b & 040000; b_{16} = b & 0100000; b_{17} = b & 0200000; b_{18} = b & 0400000; b_{19} = b & 01000000; b_{20} = b & 02000000; b_{21} = b & 04000000; b_{22} = b & 010000000; b_{23} = b & 020000000; b_{24} = b & 040000000; b_{25} = =b & 0100000000; b_{26} = b & 0200000000; b_{27} = b & 0400000000; b_{28} = b & 01000000000; b_{29} = =b & 02000000000; b_{30} = b & 04000000000; b_{31} = b & 010000000000; b_{32} = b & 020000000000, де & – операція арифметичного “І”.

7. Виконується циклічне заміщення $m + 1$ розрядів числа b за схемою: $k = b_{m+1}$, $b_{m+1} = b_m$, …, $b_2 = b_1$, $b_1 = k$.

8. З використанням бінарної операції ^ - порозрядного виключеного “АБО” – будується число $z_{i+1,j} = b \wedge Y$.

9. Дешифрованим є зображення після кроків 5 – 8.

При $P = 23$, $Q = 83$ результати наведені на рис. 1–3.

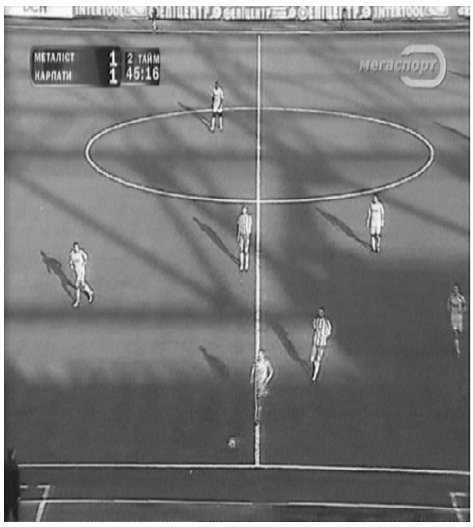

Рис. 1. Початкове зображення

Рис. 2. Зашифроване зображення

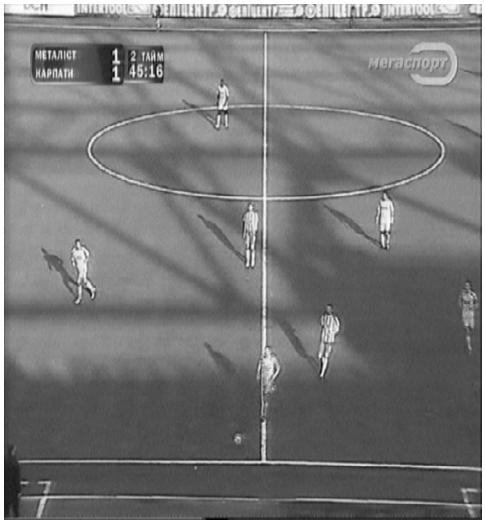

Рис. 3. Дешифроване зображення

При ***P* = 83, *Q* = 53** результати наведено на рис. 4–6.

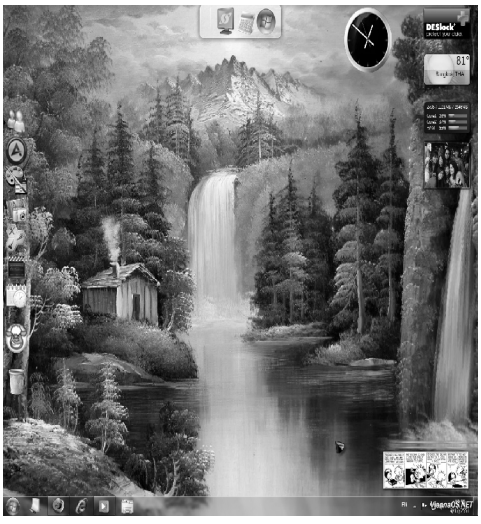

Рис. 4. Початкове зображення

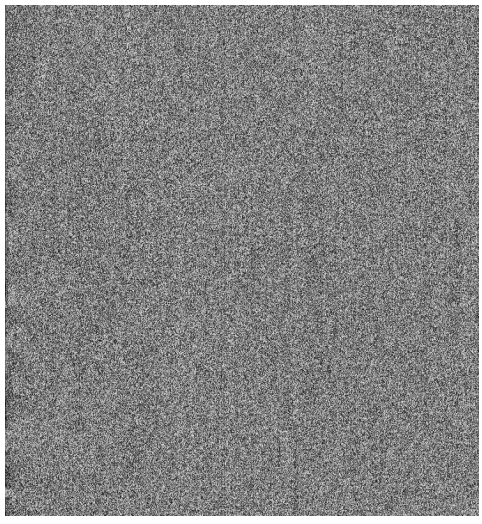

Рис. 5. Зашифроване зображення

Рис. 6. Дешифроване зображення

Порівнюючи рис. 2 і 5, бачимо, що шифрування за різних значень простих чисел ***P*** і ***Q*** суттєво відрізняється. Контури в обох зашифрованих зображеннях відсутні. Початкові і дешифровані зображення лише незначно відрізняються за рівнем яскравості.

Висновки

1. Запропонована модифікація шифрування призначена для шифрування кольорових зображень і грунтується на використанні ідей базового алгоритму RSA.
2. Запропоновану модифікацію можна використати стосовно будь-якого типу зображень, але найбільших переваг досягають у випадку використання зображень, які дають змогу чітко виокремити контури.
3. Обидва типи модифікацій без жодних застережень можна використати і стосовно монохромних зображень. Однак, незалежно від типу зображення, пропорційно до розмірності вхідного зображення зростає розмір шифрованого зображення.
4. Стійкість до несанкціонованого дешифрування запропонованою потоковою модифікацією забезпечується алгоритмом RSA і сумісним вкористанням побітових операцій.

1. Павлидис Т. Алгоритмы машиной графики и обработки изображений. – М.: Радио и связь, 1986. – 399 с. 2. Б. Яне. Цифровая обработка изображений. – М., Техносфера, 2007. – 583 с. 3. Брюс Шнайер. Прикладная криптография. – М.: Триумф, 2003. – 815 с. 4. Рашкевич Ю. М., Пелешко Д. Д., Ковальчук А. М. , Пелешко М. З. . Модифікація алгоритму RSA для деяких класів зображень // Технічні вісті. – 2008/1(27), 2(28). – С. 59–62. 5. Rashkevych Y. , Kovalchuk A. , Peleshko D. , Kupchak M. Stream Modification of RSA Algorithm For Image Coding with precize contour extraction. Proceedings of the X-th International Conference CADSM 2009. 24–28 February 2009, Lviv-Polyana, Ukraine, P. 469–473. 6. Ковальчук А. М., Попадинець К. С. Бінарні перетворення з елементами алгоритму RSA у захисті зображень за додаткового зашумлення // Вісник нац. ун-ту "Львівська політехніка" "Комп'ютерні науки та інформаційні технології". – № 843. – С. 79–84.